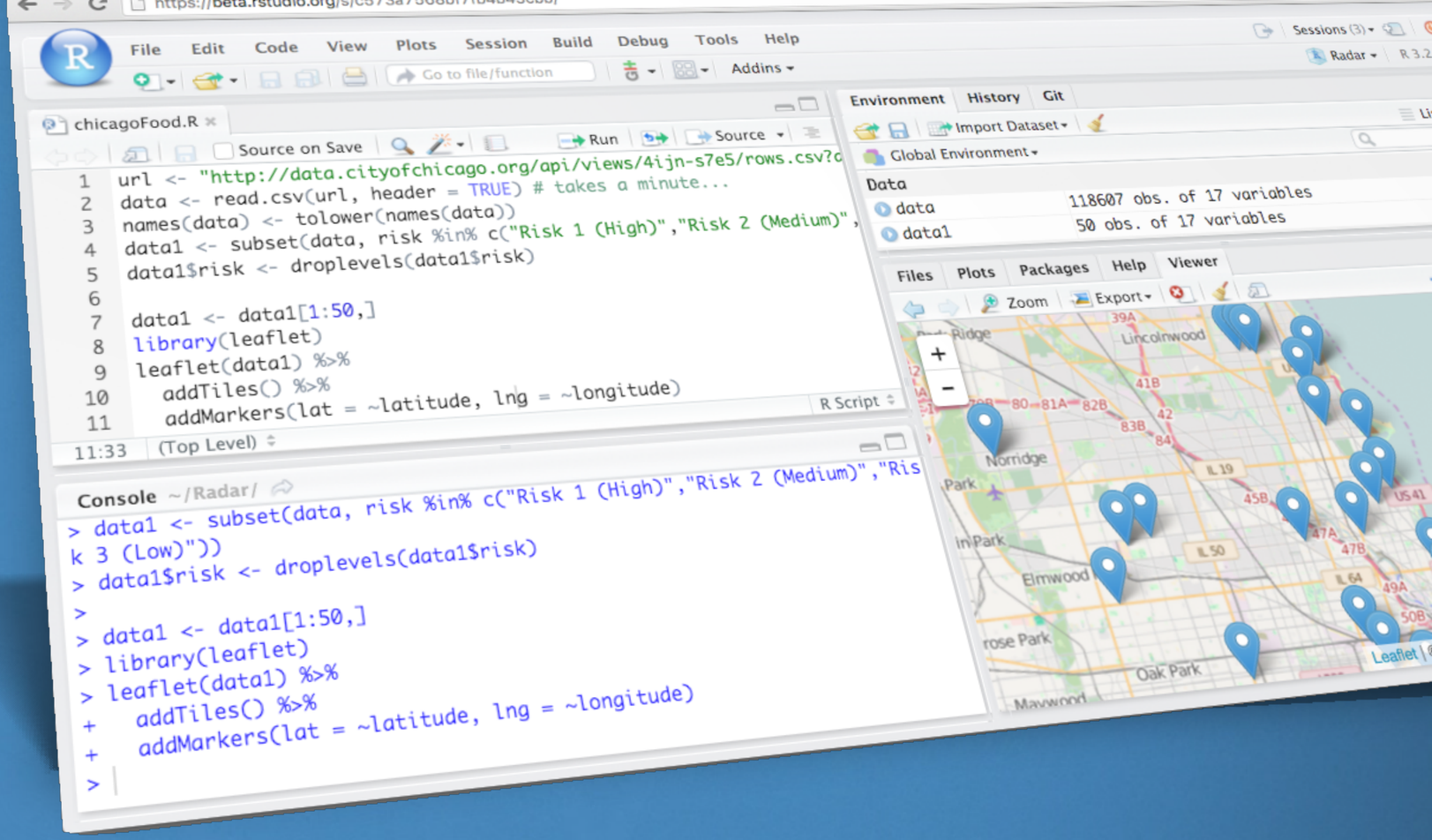
RStudio®



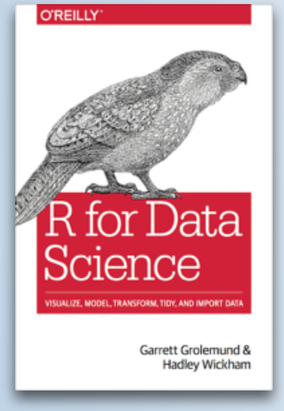
IMPORTING DATA INTO R

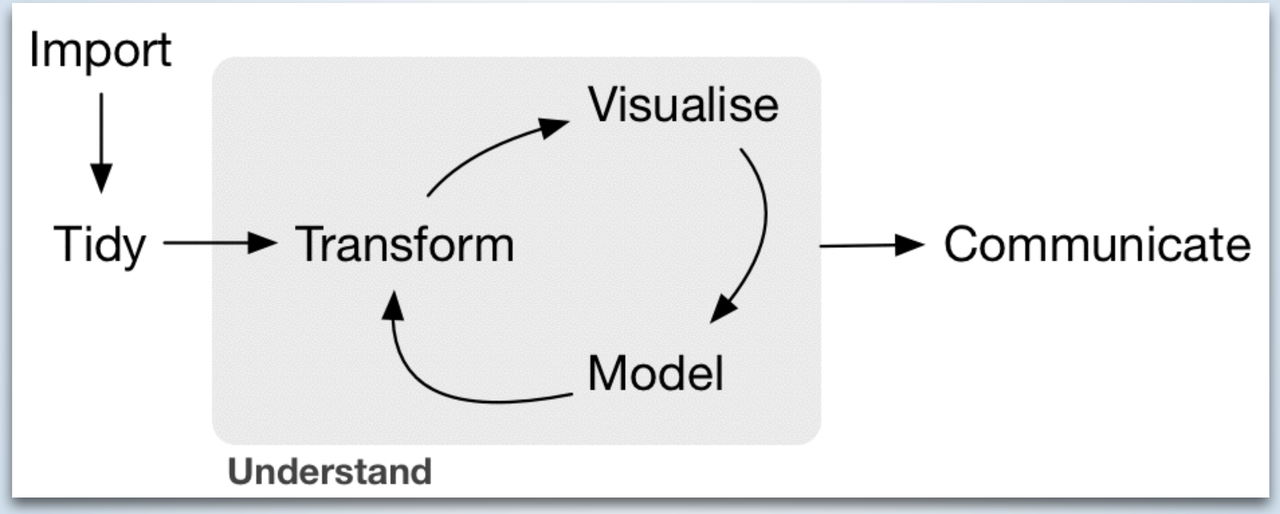
Importing Data into R

- Overview
- Importing _____ Data
  - Tabular
  - Hierarchical
  - Relational
  - Distributed
- Questions

RStudio®

Overview: The typical data science project

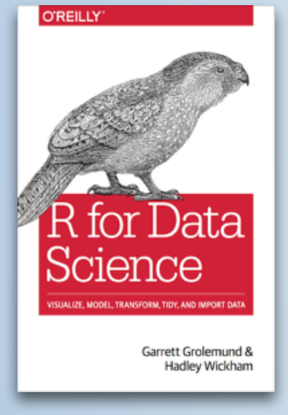


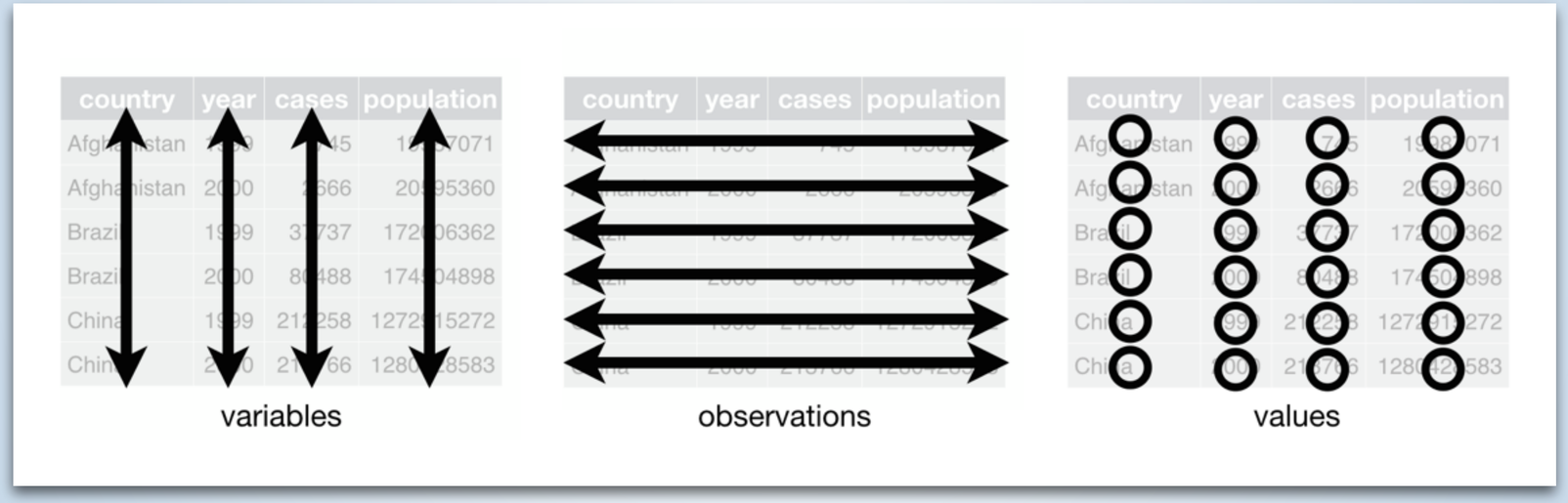


RStudio

Importing Tabular Data

What is Tabular Data?



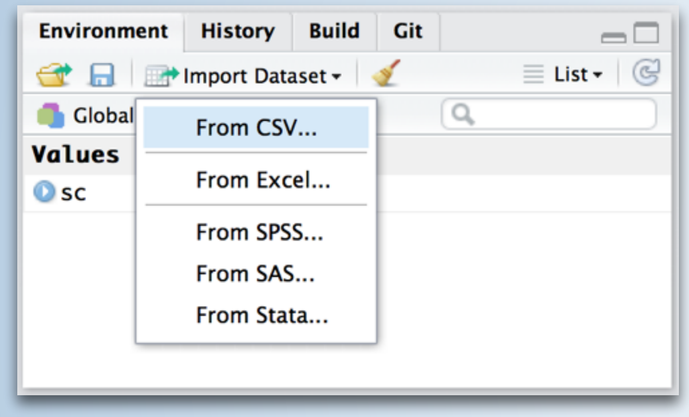


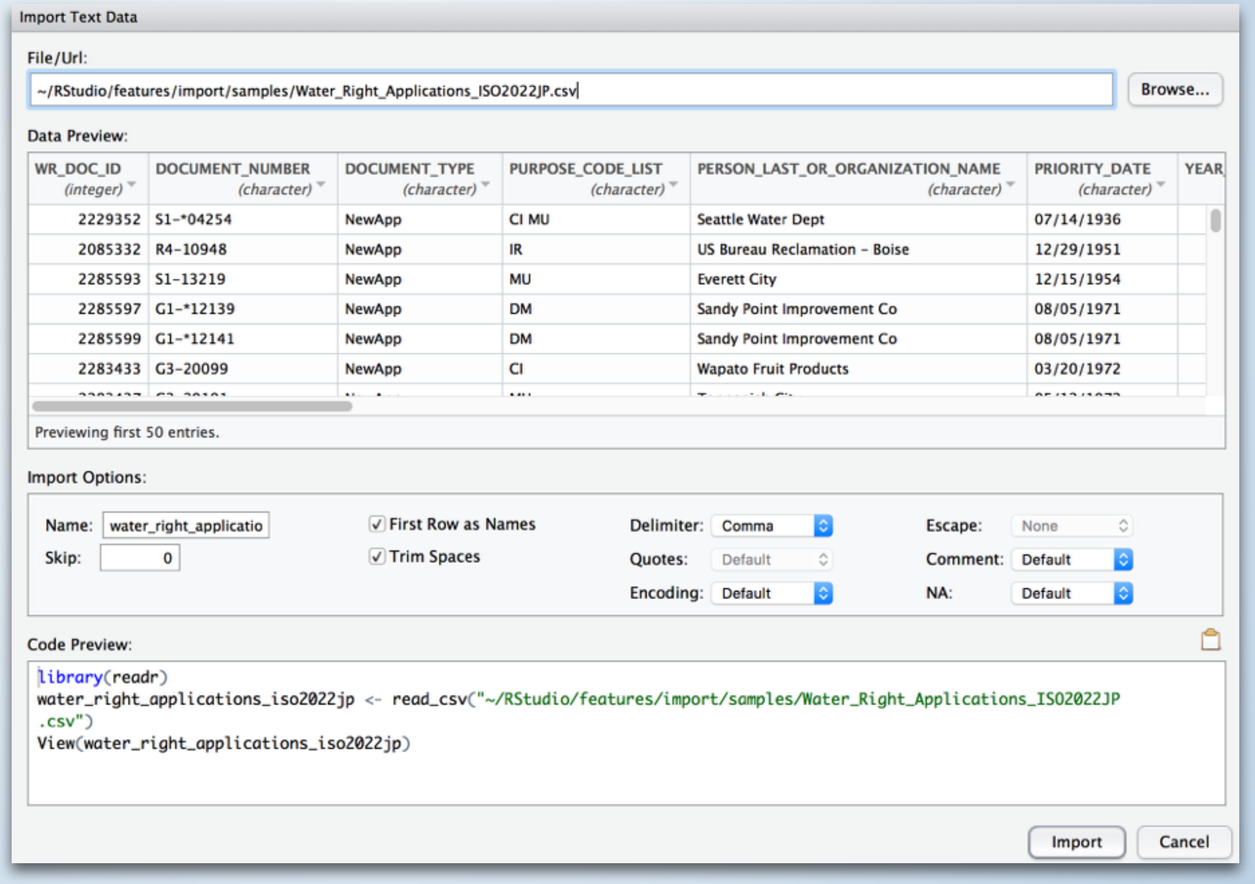
Text, Excel, SPSS, SAS and Stata

```
# Import from Text
library(readr)
read_csv("inst/data/Water_Right_Applications.csv")

# Import from Excel
library(readxl)
read_excel("inst/data/Water_Right_Applications.xls")

# Import from SPSS
library(haven)
read_sav("inst/data/Child_Data.sav")

# Import fomr SAS
read_sas("inst/data/iris.sas7bdat")

# Import from STATA
read_dta("inst/data/Milk_Production.dta")
```

RStudio

TABULAR DATA FROM RSTUDIO



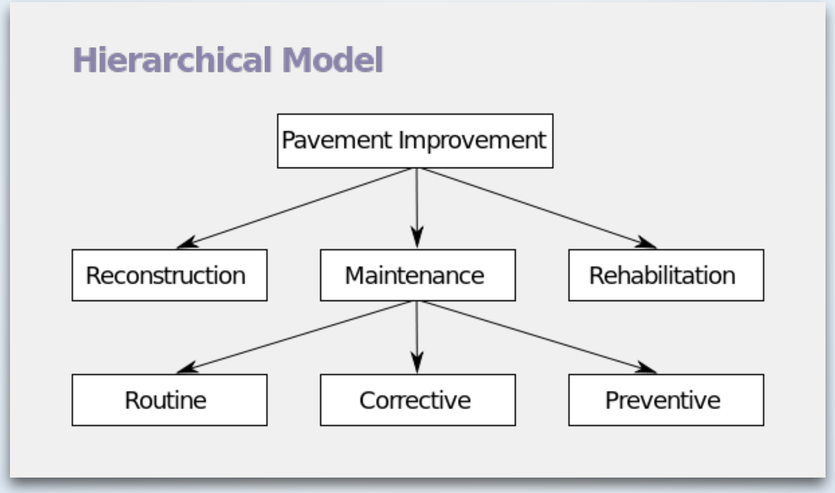


RStudio

Importing Hierarchical Data

What is Hierarchical Data?

“A data model in which the data is organized into a tree-like structure” - Wikipedia



RStudio

What is XML, HTML and JSON?

XML: Extensible Markup Language

```
<note>
  <to>Tove</to>
  <from>Jani</from>
  <heading>Reminder</heading>
  <body>Don't forget me this weekend!</body>
</note>
```

HTML: HyperText Markup Language

```
<!DOCTYPE html>
<html>
  <head>
    <title>Page Title</title>
  </head>
  <body>
    <h1>This is a Heading</h1>
    <p>This is a paragraph.</p>
  </body>
</html>
```

JSON: JavaScript Object Notation

```
{"employees":[
    {"firstName":"John", "lastName":"Doe"},
    {"firstName":"Anna", "lastName":"Smith"},
    {"firstName":"Peter", "lastName":"Jones"}
]}
```

Source: http://www.w3schools.com/

RStudio

IMPORTING XML, HTML AND JSON

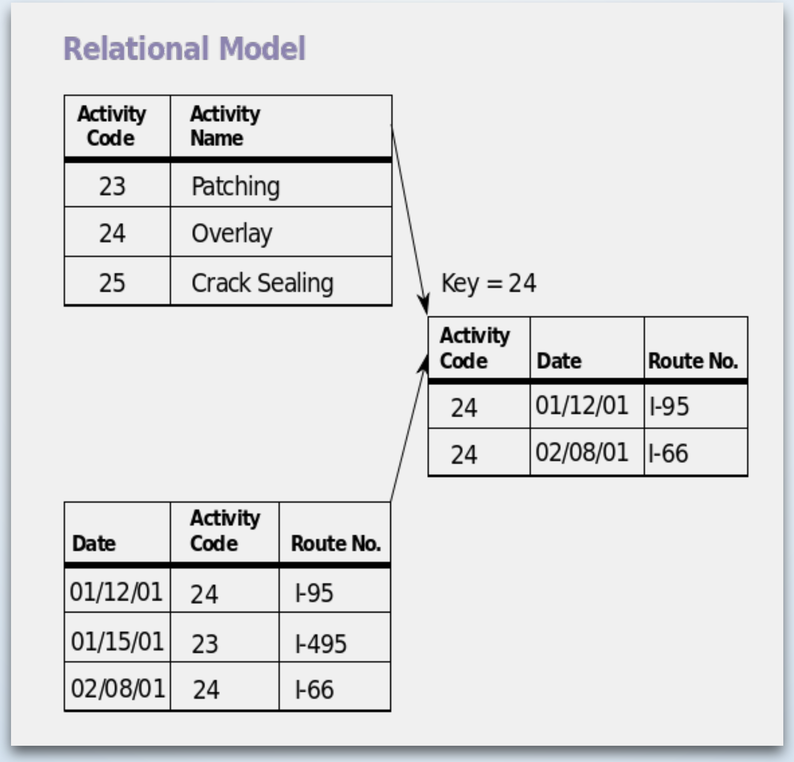
```r
# Import from XML
library(xml2)
xml <- read_xml("inst/data/Water_Right_Applications.xml")
xml_children(xml_children(xml))

# Import from HTML
library(rvest)
html <- read_html("https://en.wikipedia.org/wiki/List_of_countries_and_dependencies_by_population")
table <- xml_find_one(html, "//table")
html_table(table)

# Import from JSON
library(jsonlite)
json <- fromJSON("inst/data/Water_Right_Applications.json")
```

RStudio®

Importing Relational Data

What is Relational Data?

"Data is represented in terms of tuples, grouped into relations. " - Wikipedia



RStudio

Importing Relational Data

List of Software [edit]

- 4th Dimension
- Adabas D
- Alpha Five
- Apache Derby
- Aster Data
- Amazon Aurora
- Altibase
- CA Datacom
- CA IDMS
- Clarion
- Clustrix
- CSQL
- CUBRID
- DataEase
- Database Management Library
- Dataphor
- dBase
- Derby aka Java DB
- Empress Embedded Database
- EXASolution
- EnterpriseDB
- eXtremeDB
- FileMaker Pro
- Firebird
- FrontBase
- Greenplum
- GroveSite
- Hadoop
- H2
- Helix database
- HSQLDB
- IBM DB2
- IBM Lotus Approach
- IBM DB2 Express-C
- Infobright
- Informix
- Ingres
- InterBase
- InterSystems Caché
- GT.M
- LibreOffice Base
- Linter
- MariaDB
- MaxDB
- MemSQL
- Microsoft Access
- Microsoft Jet Database Engine (part of Microsoft Access)
- Microsoft SQL Server
- Microsoft SQL Server Express
- SQL Azure (Cloud SQL Server)
- Microsoft Visual FoxPro
- Mimer SQL
- MonetDB
- mSQL
- MySQL
- Netezza
- NexusDB
- NonStop SQL
- NuoDB
- Openbase
- OpenLink Virtuoso (Open Source Edition)
- OpenLink Virtuoso Universal Server
- OpenOffice.org Base
- Oracle
- Oracle Rdb for OpenVMS
- Panorama
- Pervasive PSQL
- Polyhedra
- PostgreSQL
- Postgres Plus Advanced Server
- Progress Software
- RDM Embedded
- RDM Server
- R:Base
- The SAS system
- SAND CDBMS
- SAP HANA
- SAP Sybase Adaptive Server Enterprise
- SAP Sybase IQ
- SQL Anywhere (formerly known as Sybase Adaptive Server Anywhere and Watcom SQL)
- ScimoreDB
- SmallSQL
- solidDB
- SQLBase
- SQLite
- Sybase Advantage Database Server
- Teradata
- TimesTen
- Trafodion
- txtSQL
- Unisys RDMS 2200
- UnQLite
- UniData
- UniVerse
- Vectorwise
- Vertica
- VMDS

Icons Credit: Hans Gerhard Meier and Wilson Joseph

RStudio

Importing Relational Data

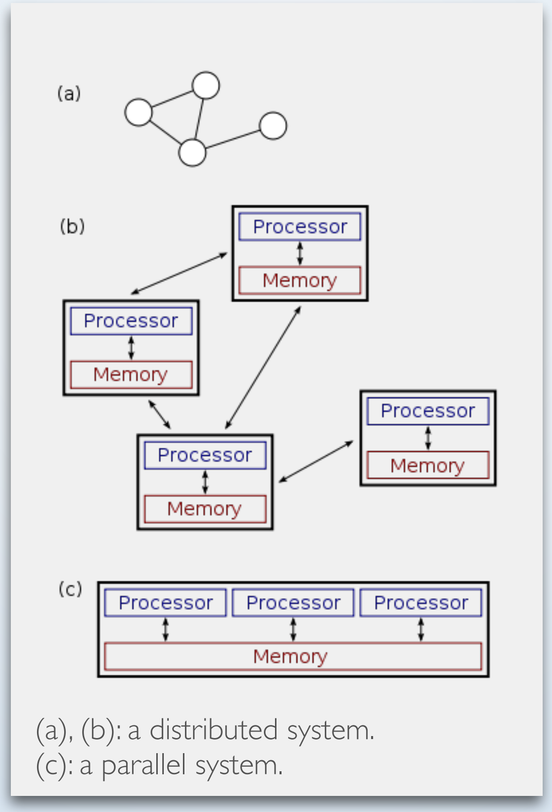
```
# Connect to Postgres
library(DBI)
db <- dbConnect(RPostgres::Postgres(), user, pass, ...)

# Connect to MySQL
db <- dbConnect(RMySQL::MySQL(), user, pass, ...)

# Connect to SQLite
db <- dbConnect(RSQLite::SQLite(), dbname = "inst/data/database.sqlite")

# Import data from SQLite
dbListTables(db)
View(dbGetQuery(db, "SELECT * FROM packages"))

# Disconnect
dbDisconnect(db)
```

RStudio

Importing Distributed Data

What is Distributed Data?

“non-relational with quick access to data over a large number of nodes” - Wikipedia

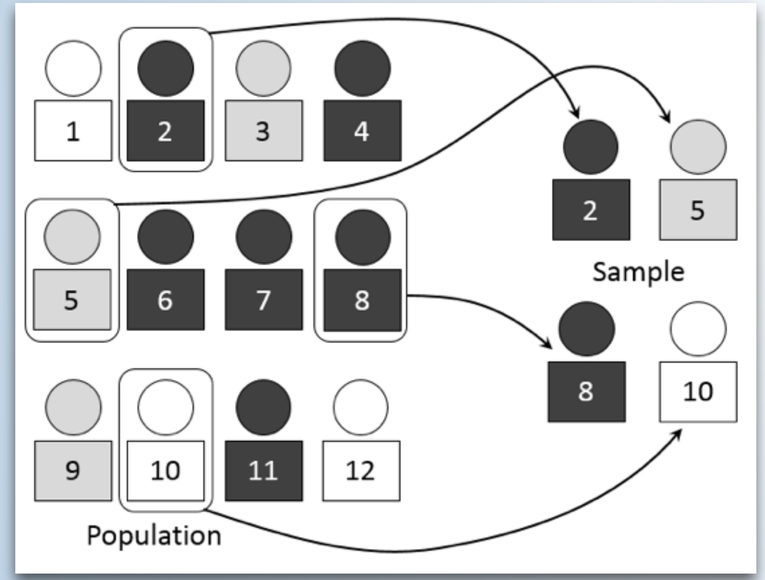


(a), (b): a distributed system.
(c): a parallel system.

RStudio®

Importing Distributed Data

Sampling



Distributed computing

RStudio


RStudio®